

## **IN THE CLAIMS**

1. (Previously Presented) A method for speculatively reusing regions of code, the method comprising:
  - identifying a reuse region and input data of the reuse region;
  - searching a plurality of instances of the reuse region for a valid matching instance that has matching input data and cannot be potentially invalidated; and
  - if the valid matching instance is not found, predicting, for the reuse region, a current set of live-out values using reuse region instance information pertaining to the plurality of instances of the reuse region.
2. (Currently Amended) The method of claim 1 wherein searching [[the]] the plurality of instances comprises:
  - determining whether the input data of the reuse region matches any input information within the reuse region instance information; and
  - if the input data matches input information within the reuse region instance information, determining whether the reuse region is identified by a normal reuse instruction.
3. (Original) The method of claim 1 wherein the reuse region instance information includes input information and output information for each instance of the reuse region.
4. (Previously Presented) The method of claim 3 wherein the reuse region instance information further includes a plurality of confidence counters for each live-out register

of the reuse region, each of the plurality of confidence counters being associated with a prediction technique.

5. Canceled

6. (Previously Presented) The method of claim 4 wherein predicting the current set of live-out values comprises:

comparing a plurality of confidence counts associated with each live-out value in the set;

selecting an optimal prediction technique for said each live-out value in the set based on comparison; and

selecting an output value for said each live-out value in the set using the optimal prediction technique.

7. (Previously Presented) The method of claim 6 wherein the optimal prediction technique is any one of a context-based prediction technique, a stride prediction technique, and a last value prediction technique.

8. (Previously Presented) An apparatus comprising:

a buffer to hold reuse region instance information pertaining to a plurality of instances of a reuse region; and

a processing core to search the buffer for a valid matching instance that has input data matching current input data of the reuse region and cannot be potentially invalidated, to predict for the reuse region a current set of live-out registers and an output value for

each live-out register in the set based on the reuse region instance information if the valid matching instance is not found in the buffer, and to speculatively execute instructions using predicted output values of the reuse region.

9. Canceled.

10. (Currently Amended) The apparatus of claim 8[[9]] wherein the processing core is to search the buffer for a valid matching instance by finding an instance with input data matching the current input data of the reuse region and determining whether the reuse region is identified by a normal reuse instruction.

11. (Original) The apparatus of claim 8 wherein the reuse region instance information includes input information and output information for each instance of the reuse region.

12. (Previously Presented) The apparatus of claim 11 wherein the reuse region instance information further includes a plurality of confidence counters for each live-out register of the reuse region, each of the plurality of confidence counters being associated with a prediction technique.

13. (Previously Presented) The apparatus of claim 8 wherein the buffer includes a prediction list having a plurality of pointers to reuse region instances held in the buffer, a pointer to the most currently used instance being located at the top of the prediction list and a pointer to the least currently used instance being located at the bottom of the prediction list.

14. (Original) The apparatus of claim 8 wherein the buffer includes a value prediction table having an entry that includes a predicted output value, the predicted output value being located using an index.

15. (Previously Presented) The apparatus of claim 13 wherein the processing core is to predict an output value for each live-out register by comparing a plurality of confidence counters associated with said each live-out register, selecting an optimal prediction technique for said each live-out register based on comparison, and selecting the output value for said each live-out register using the optimal prediction technique.

16. Canceled.

17. (Previously Presented) The apparatus of claim 13 wherein:

the prediction list points to a most recently used instance if the optimal prediction technique is a last value prediction technique;

the prediction list points to two most recently used instances if the optimal prediction technique is a stride prediction technique; and

the prediction list points to instances associated with a corresponding live-out register if the optimal prediction technique is a context-based prediction technique, the associated instances being used to calculate an index pointing to a predicted output value in a value prediction table maintained in the buffer.

18. (Previously Presented) A system comprising:

a memory to store regions of code; and

a processor, coupled to the memory, to identify a reuse region in the regions of code, to search a plurality of instances of the reuse region for a valid matching instance that has matching input data and cannot be potentially invalidated, and to predict for the reuse region a current set of live-out values using reuse region instance information pertaining to the plurality of instances of the reuse region if the valid matching instance is not found.

19. Canceled.

20. (Previously Presented) The system of claim 18 wherein the reuse region instance information includes input information and output information for each instance of the reuse region.

21. (Previously Presented) The system of claim 18 wherein the reuse region instance information includes a plurality of confidence counters associated with each live-out value in the set, each of the plurality of confidence counters being associated with a prediction technique.

22. Canceled.

23. Canceled.

24. (Previously Presented) A machine accessible medium comprising instructions, which when executed on a processing system, perform a method for speculatively reusing regions of code, the method comprising:

identifying a reuse region and input data of the reuse region;

searching a plurality of instances of the reuse region for a valid matching instance that has matching input data and cannot be potentially invalidated; and

if the valid matching instance is not found, predicting, for the reuse region, a current set of live-out values using reuse region instance information pertaining to the plurality of instances of the reuse region.

25. (Previously Presented) The machine accessible medium of claim 24 wherein searching the plurality of instances of the reuse region comprises:

determining whether the input data of the reuse region matches any input information within the reuse region instance information; and

if the input data matches input information within the reuse region instance information, determining whether the reuse region is identified by a normal reuse instruction.

26. (Previously Presented) The machine accessible medium of claim 24 wherein the reuse region instance information includes input information and output information for each instance of the reuse region.

27. (Previously Presented) The machine accessible medium of claim 26 wherein the reuse region instance information further includes a plurality of confidence counters

associated with each live-out value in the set, each of the plurality of confidence counters being associated with a prediction technique.

28. Canceled.

29. (Previously Presented) The machine accessible medium of claim 27 wherein predicting the current set of output values comprises :

comparing the plurality of confidence counters;

selecting an optimal prediction technique for said each live-out value in the set based on comparison; and

selecting an output value for said each live-out value in the set using the optimal prediction technique.

30. (Previously Presented) The machine accessible medium of claim 29 wherein the optimal prediction technique is any one of a context-based prediction technique, a stride prediction technique, and a last value prediction technique.